(ii), and (iii);

b) during the fabrication of paragraph (a),

i) fabricating identical CONTROL modules in all systems;

a'

ii) fabricating identical COM_MOD modules in all systems;

iii) fabricating PAK_MOD modules in all systems, such that:

A) copies of a software unit A is contained in every PAK_MOD module;

B) some PAK_MOD modules contain a software unit B with no unit C; and

C) some PAK_MOD modules contain a software unit C with no unit B.

REMARKS

This Amendment is submitted in response to the Office Action mailed on May 23, 2002. Claims 1 - 7 are pending, and all stand rejected at present, on grounds of anticipation, based on Yates.

The amendment to claim 6 corrects an obvious typographical error, namely, the word "come" in claim 6(b)(iii)(C).

Claim 1

Claim 1 recites:

1. A method of constructing a plurality of software systems, comprising the following steps:

a) maintaining an inventory of software modules, which includes:

i) a group of type A modules; and

ii) a collection of type B modules;

b) when constructing each software system,

i) including copies of the entire group of type A modules;

ii) including copies of selected type B modules; and

iii) generating at least one customized module, which is a copy of neither a type A nor a Type B module.

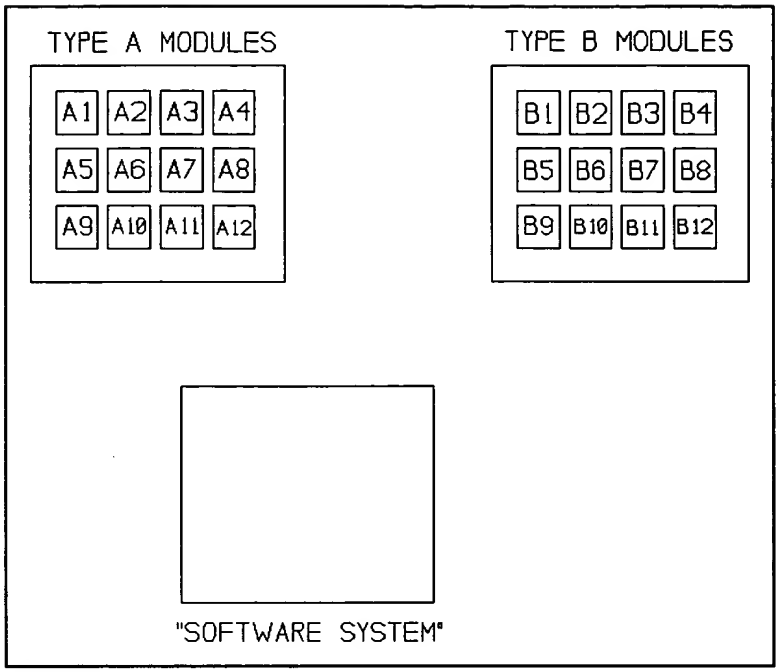
Graphical Illustration of Claim 1

Sketches 1 and 2, below, graphically illustrate some concepts behind claim 1. Sketch 1 illustrates

1) "a group of type A modules" [see claim 1(a)(i)],

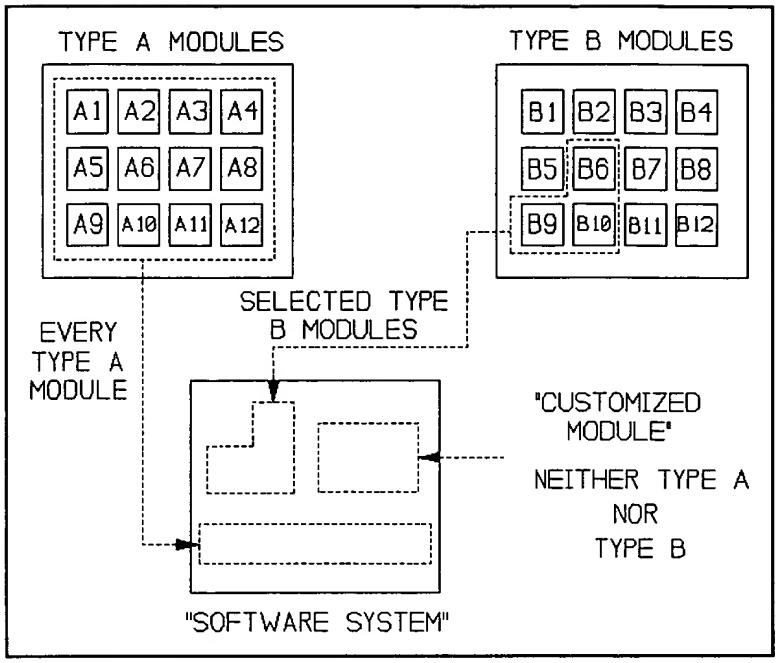
2) "a collection of type B modules" [see claim 1(a)(ii), and

3) a "software system" to be constructed [see claim 1(b)].



SKETCH 1

SKETCH 2 shows construction of the "software system."



SKETCH 2

Applicant's Comments

Applicant points out the following significant features of SKETCH 2.

1) **ALL** type A modules are copied into the "software system." [Claim 1(b)(i).]

2) **SELECTED** type B modules are copied into the "software system." [Claim 1(b)(ii).]

3) A **CUSTOMIZED** module is generated, which contains neither a copy of type A nor type B.

Office Action's Rejection

The Office Action relies on Yates, column 18, lines 1 - 13 to show claim 1(b)(i) and (b)(ii). That passage of Yates is here set forth:

> Known constructions, where policies are embedded in the objects, require rewriting of code in the object to change behavior.
>
> External policies allow not only changes in behavior to be achieved more easily but also more freely, and can allow extra behaviors (which are composed from combinations/permutations of a programmed set of operations) to be performed even if these were not originally anticipated.
>
> The concept of policies is such that an object must have access to a "Policy Interpreter."
>
> This can be internal or external to the object.
>
> In order to to locate policies, a policy server might be provided, again either internal or external to an object.
>
> (Yates, column 18, lines 1 - 13.)

Plainly, the Yates passage has no relevance to claim 1(b)(i) and (b)(ii).

MPEP § 2131 states:

> A claim is anticipated only if **each and every element** as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.

Applicant submits that claim 1(b)(i) and (b)(ii) have not been shown in Yates, and thus claim 1 cannot be anticipated.

In addition, Applicant points out that it is axiomatic that, in order to **anticipate** claim 1, the Yates reference must **infringe** claim 1. (See PATENTS, A Treatise on the Law of Patentability, Validity, and Infringement, by D. Chisum, section 3.02[1], entitled, "The Classic Infringement Test.")

Applicant submits that the Yates passage set out above clearly does not show claim 1(b)(i) and (b)(ii).

Therefore, Applicant submits that Yates does not anticipate claim 1.

Claims 2 - 5

Claims 2 - 5 depend from claim 1, and are considered allowable based on claim 1.

Claim 6

Claim 6 recites:

> 6. An expedited method of assembling a software system, comprising the following steps:
>
> a) fabricating a collection of software systems, each of which contains
>
> i) a processing module (PROC_MOD) which processes content of messages;

ii) a packaging module (PAK_MOD) which packages messages into packets for transport out of the system;

iii) a communication module (COM_MOD) which accepts and delivers message packets; and

iv) a system control module (CONTROL) which coordinates the processes of (i), (ii), and (iii);

b) during the fabrication of paragraph (a),

i) fabricating identical CONTROL modules in all systems;

ii) fabricating identical COM_MOD modules in all systems;

iii) fabricating PAK_MOD modules in all systems, such that:

A) copies of a software unit A is contained in every PAK_MOD module;

B) some PAK_MOD modules contain a software unit B with no unit C; and

C) some PAK_MOD modules contain a software unit C with no unit B.

Applicant points out that claim 6(b) recites:

a) fabricating a collection of software systems, each of which contains

and then lists four types of module. Restated, claim 6(b) states that every "software system" in the "collection" contains (at least) the four modules listed in 6(a)(i) through (a)(iv).

The Office Action cites two passages in Yates to show this, namely,

1) column 2, lines 38 - 65

and

2) column 4, lines 13 - 65.

However, those passages contain nothing more than generalized statements indicating that, in different situations, systems may be designed which are different.

That is **directly contrary** to the claim recitations in question.

Applicant requests, under 37 CFR §§ 1.104(c)(2) and 35 U.S.C. § 132, that the PTO specifically identify a statement in Yates that each software module in a "collection" must contain the four elements of Applicant's claim 1(a)(i) through (a)(iv).

Further, claim 6(b) states that all of the "software systems" of claim 6(a) will contain two elements, namely (1) identical CONTROL modules and (2) identical COM_MOD modules. The Office Action relies on two passages in Yates to show this.

One passage is column 4, lines 26 -35. However, that passage merely states that "at least one . . . software agent" is equipped with certain functionality. That does not show the claim recitations in question, and is actually inconsistent with it.

It is inconsistent because the passage only focuses on **ONE** software agent, and lists some properties of that agent. It fails

to identify a **GROUP** of agents. Claim 6 refers to properties of a "collection of software systems." Yates' discussion of a **SINGLE** software agent does not show the properties of a **GROUP** as in claim 6.

The other passage relied on by the PTO is Yates column 18, lines 1 - 13. That passage is set out verbatim above, and clearly does not show claim 6(b)(i) and (b)(ii).

Applicant points to claim 6(b)(iii), which is repeated here:

> iii) fabricating PAK_MOD modules in all systems, such that:
>
> > A) copies of a software unit A is contained in every PAK_MOD module;
> >
> > B) some PAK_MOD modules contain a software unit B with no unit C; and
> >
> > C) some PAK_MOD modules contain a software unit C with no unit B.

The undersigned attorney has examined the passages in Yates which are cited to show these recitations, and cannot locate the recitations in those passages. [Actually, the passages used by the PTO are the same as used for claim 6(b)(i) and (b)(ii).]

Therefore, Applicant specifically requests that claim 6(b)(iii) be identified in Yates. Applicant points out that this claim passage recites characteristics of certain PAK_MOD modules:

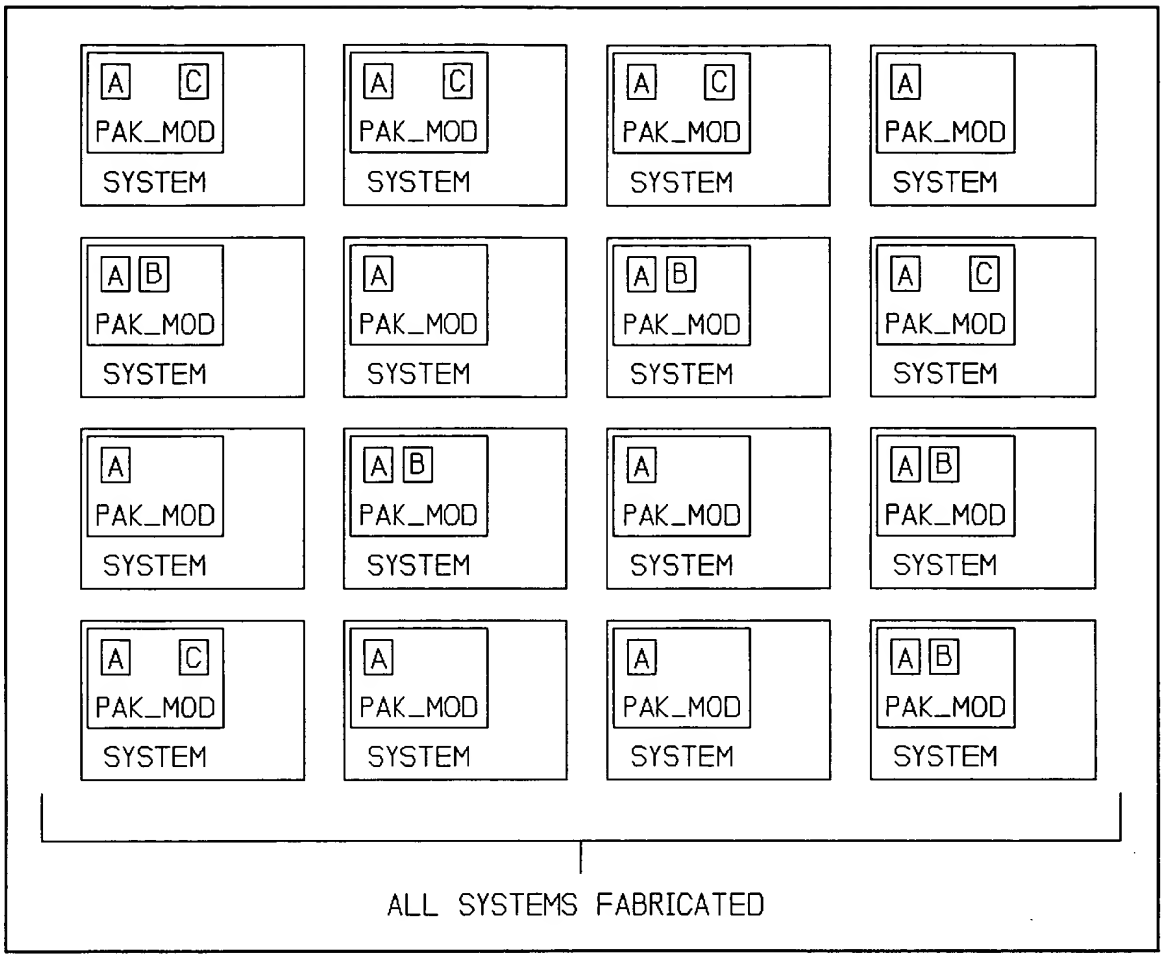
-- All contain a "software unit A;"

-- Some contain a unit B, but no C;

and

-- Some contain C, but no B.

Applicant requests that this particular combination of elements be identified in Yates. Sketch 3, below, provides one illustration of the combination, and is considered self-explanatory. That combination has not been shown in Yates.



SKETCH 3

Applicant points out specific recitations of claim 6 which SKETCH 3 illustrates:

-- Every system contains a PAK_MOD module [claim 6(b)(iii)];

-- Every PAK_MOD module contains a unit A [claim 6(b)(iii)(A)];

-- Some PAK_MOD modules contain unit B with no unit C [claim 6(b)(iii)(B)];

-- Some PAK_MOD modules contain unit C with no unit B [claim 6(b)(iii)(C)].

That combination has not been shown in Yates.

In addition, Applicant points out that claim 6(b)(i) and (b)(ii) recites that all "systems" contain **IDENTICAL** CONTROL and COM_MOD modules. (That is not shown in SKETCH 3.)

Claim 7

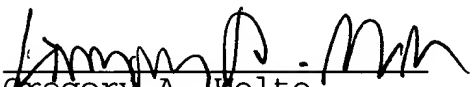
Claim 7 depends from claim 6, and is considered allowable based on claim 6.

Conclusion

Applicant requests that the rejections to the claims be reconsidered and withdrawn.

Applicant expresses thanks to the Examiner for the careful consideration given to this case.

Respectfully submitted,



Gregory A. Welte
Reg. No. 30,434

NCR Corporation
1700 South Patterson Blvd.
WHQ - 5E
Dayton, OH 45479
October 21, 2002
(937) 445 - 4956
ATTACHMENT: Annotated Claim(s) Showing Amendments

ATTACHMENT: Annotated Claim(s) Showing Amendments

6. (Amended once.) An expedited method of assembling a software system, comprising the following steps:

a) fabricating a collection of software systems, each of which contains

i) a processing module (PROC_MOD) which processes content of messages;

ii) a packaging module (PAK_MOD) which packages messages into packets for transport out of the system;

iii) a communication module (COM_MOD) which accepts and delivers message packets; and

iv) a system control module (CONTROL) which coordinates the processes of (i), (ii), and (iii);

b) during the fabrication of paragraph (a),

i) fabricating identical CONTROL modules in all systems;

ii) fabricating identical COM_MOD modules in all systems;

iii) fabricating PAK_MOD modules in all systems, such that:

A) copies of a software unit A is contained in every PAK_MOD module;

B) some PAK_MOD modules contain a software unit B with no unit C; and

C) [come] some PAK_MOD modules contain a software unit C with no unit B.